

S/N 10/765,556

Amendments of the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application.

Please amend claims 14, 16 – 24, 26, 27, 33, 41 and 42, as follows:

5 **Listing of claims.**

1. (Original) A digital processing system comprising:

a first semiconductor device having a plurality of computation nodes;

a second semiconductor device having a microprocessor-based node adapted to function as a system controller and a plurality of computation nodes; and

10 a bus for inter-connecting said first semiconductor device to said second semiconductor device in a ring topology whereby the microprocessor-based node allocates a plurality of operations among said computation nodes of said first and second semiconductor devices over said inter-connecting bus and receives the results of said operations over said inter-connecting bus.

15

2. (Original) The digital processing system of claim 1 wherein said computation nodes on said first and second semiconductor devices include at least one of the following node types: an arithmetic node, a bit manipulation node, a reduced instruction set processing node and a finite state machine node.

20

3. (Original) The digital processing system of claim 1 wherein the digital system includes a plurality of nodes each of which can perform at least one type of operation and wherein said microprocessor node allocates functions temporally and spatially among the nodes of said first and second semiconductor devices.

25

4. (Original) The digital processing system of claim 1 further comprising:

a remote processor; and a second bus for interfacing said first and second semiconductor device to said remote processor.

30

S/N 10/765,556

5. (Original) The digital processing system of claim 4 further comprising:
a real-time data source; and
a third bus for interfacing said first and second semiconductor device to said real-time data generator.

5

6. (Original) The inter-connecting bus of the digital processing system of claim 1 further comprising:

a plurality of logic elements, each closely proximate to a corresponding output pad of said first semiconductor device, for sending data and control signals;

10 a plurality of logic elements, each closely proximate to a corresponding input pad of said second semiconductor device, for receiving said data and control signals;

and means for synchronously transferring data and control logic by clocking the logic elements of said first and second semiconductor devices.

15 7. (Original) The digital processing system of claim 6 further comprising:

a first core logic clock signal provided to each of said nodes of said first semiconductor device;

a second core logic clock signal provided to each of said nodes of said second semiconductor device, said first and second core logic clocks having substantially the same frequency and phase;

20 a first bus clock for clocking said logic elements of said first semiconductor device at a selected rate where said first bus clock is derived from said first core logic clock;

a second bus clock for clocking said logic elements of said second semiconductor device at said selected rate where said second bus clock is derived from said second core logic clock; and

25 a synchronizing signal generated by said first semiconductor device.

8. (Original) The digital processing system of claim 1 further comprising:

30 a first core logic clock signal provided to each of said nodes of said first semiconductor device;

S/N 10/765,556

a second core logic clock signal provided to each of said nodes of said second semiconductor device, said first and second core logic clocks having substantially the same frequency and phase; and

means for synchronizing the transfer for data and control signals from said first semiconductor device to said second semiconductor device, said synchronizing means
5 derived from said first core logic clock signal.

9. (Original) The synchronizing means of claim 8 further comprising:

a first bus clock for clocking said logic elements of said first semiconductor
10 device at a selected rate where said first bus clock is derived from said first core logic clock;

a second bus clock for clocking said logic elements of said second semiconductor device at said selected rate where said second bus clock is derived from said second core logic clock; and

15 a signal, generated by said first semiconductor device, for indicating valid data and control signals to said second semiconductor device.

10. (Original) The digital processing system of claim 1 wherein said first and second semiconductor devices are coupled in a ring with said inter-connecting bus coupling an
20 output port of said first semiconductor device to an input port of said second semiconductor device and further coupling an output port of said second semiconductor device to an input port of said first semiconductor device.

11. (Original) The digital processing system of claim 1 wherein said microprocessor-
25 based node determines whether a packet received on said second bus is addressed to a node on either said first or second semiconductor device.

S/N 10/765,556

12. (Original) The digital processing system of claim 11 wherein said microprocessor-based node passes said packet to one of said plurality of computation nodes on said first semiconductor device if said packet is addressed to a node on said first semiconductor device or transferring said packet to one of said plurality of computation nodes on said second semiconductor device if said packet is addressed to a node on said second semiconductor device where said packet is transferred on said inter-connecting bus.

13. (Original) The digital processing system of claim 11 wherein said packet includes a device identification field.

14. (Currently amended) A digital processing system comprising:

a plurality of adaptive computing engines each having a plurality of computation nodes;

said adaptive computing engines coupled to each other by in a ring topology by a bus that couples a first adaptive computing engine to a second adaptive computing engine, couples said second adaptive computing engine to a third adaptive computing engine and couples said third adaptive computing engine to said first adaptive computing engine, said bus adapted for passing packets between said computation nodes by passing them around said ring topology bus;

one of said adaptive computing engines having a kernel node, said kernel node adapted for determining whether a packet is addressed intended for one of said computation nodes in one of said adaptive computing engines and for discarding packets ~~intended for~~ addressed to computational nodes that are not part of said digital processing system.

15. (Original) The digital processing system of claim 14, wherein each of said adaptive computing engines includes a device identification number and said packet includes a device identification field.

S/N 10/765,556

16. (Currently amended) The digital processing system of claim 15, wherein each adaptive computing engine includes an output port coupled to an input port of the next adjacent adaptive computing engine in said ring topology bus.

5 17. (Currently amended) The digital processing system of claim 16, wherein said ring topology bus comprises more than two, ~~three or four~~ adaptive computing engines.

18. (Currently amended) The digital processing system of claim ~~[[14]]~~ 16 wherein at least one of said output ~~port~~ ports comprises a plurality of D type flip flops coupled to a
10 corresponding plurality of D type flip flops at said input port.

S/N 10/765,556

19. (Currently amended) A method for transferring information from one adaptive computing engine having a plurality of computational nodes to another adaptive computing engine having a second plurality of computational nodes, each adaptive computing engine having an input port and an output port, said adaptive computing engines being coupled to each other through a ring topology bus, said method comprising the steps of:

~~Generating~~ generating a common clock signal routed to at least two adaptive computing engines;

~~Requesting~~ requesting to transfer information from a first one adaptive computing engine to another a second adaptive computing engine via said ring topology bus;

~~Offering~~ offering to receive said information;

~~Transferring~~ transferring said information from a D type flip flop associated with an output port of a first adaptive computing engine ~~said output port~~ to a D type flip flop associated with an input port of a second adaptive computing engine via said ring topology bus; ~~said input port;~~

~~Controlling~~ controlling the transfer of said information from said D type flip flop associated with said output port of said first adaptive computing engine with a first clock derived from said common clock signal;

~~Controlling~~ controlling the receipt of said information by said D type flip flop associated with said input port of said second adaptive computing engine with a second clock derived from said common clock signal; and

~~Generating~~ generating a signal to define a time period ~~window~~ during which the transfer of said information is valid.

20. (Currently amended) The method of claim 19 further comprising the step of coupling up to four adaptive computing engines in a said ring topology bus.

S/N 10/765,556

21. (Currently amended) The method of claim 20 further comprising the steps of:
designating one of said adaptive computing engines in said ring topology bus to
function as a kernel node;
initially receiving said information at said designated adaptive computing engine;
5 and
determining whether said information is intended for one of the adaptive
computing engines in said ring topology bus.

22. (Currently amended) The method of claim 21 further comprising the step of
10 transferring said information from adaptive computing engine to adaptive computing
engine around said ring topology bus until said information arrives at the intended
adaptive computing engine.

23. (Currently amended) The method of claim 22 further comprising the step of
15 discarding said information if said kernel node determines the information is not intended
for one of the adaptive computing engines in said ring topology bus.

24. (Currently amended) The method of claim 23 further comprising the step of passing
said information from adaptive computing engine to adaptive computing engine around
20 said ring topology bus until said information arrives at its intended destination adaptive
computing engine.

25. (Original) The method of claim 21 further comprising the step of providing a FIFO
buffer, coupled to said D type flip flop associated with said output port, for retaining
25 information until the input port offers to receive information.

26. (Currently amended) The method of claim 25 further comprising the step of
providing a FIFO buffer, coupled to said D type flip flop associated with said input port
of said second adaptive computing engine said FIFO buffer, ~~for~~ receiving information
30 after closing said window that indicates the transfer of said information is valid.

S/N 10/765,556

27. (Currently amended) A digital device comprising:

a processor;

a user interface controlled by said processor; memory associated with said

5 processor; and

a plurality of adaptive computing engine devices each having a plurality of computation nodes;

said adaptive computing engine devices coupled in a ring topology by a first bus that extends between first and second adaptive computing engine devices that are
10 adjacent to each other in said ring topology;

at least one of said adaptive computing devices coupled to said processor and said memory by a second bus so that each of said plurality of computation nodes can access said memory.

15 28. (Original) The digital device of claim 27, further comprising: a real-time data source; and a third bus coupling said real-time data source to at least one of said plurality of adaptive computing engine devices.

20 29. (Original) The digital device of claim 27, wherein each of said plurality of adaptive computing engine devices is coupled to said second bus.

30. (Original) The digital device of claim 27, wherein said first bus couples an output port of each device to an input port of a next device in said ring.

25 31. (Original) The digital device of claim 27, wherein said first bus includes:

a plurality of D type flip flops, each closely proximate to a corresponding output pad of output port, for sending data and control signals;

a plurality of D type flip flops, each closely proximate to a corresponding input pad of said output port, for receiving said data and control signals; and

30 means for synchronously transferring data and control logic by clocking the logic elements of said first and second semiconductor devices.

S/N 10/765,556

32. (Original) The digital device of claim 27, wherein one of said plurality of adaptive computing engine devices includes a kernel node.

5 33. (Currently amended) An integrated circuit comprising:

a plurality of computational elements including a plurality of arithmetic nodes, a plurality of bit-manipulation nodes, a plurality of finite state machine nodes, and a plurality of input/output nodes;

10 a first and a second processing node each having a corresponding core processor based on a common architecture;

a first memory associated with said first processing node;

a second memory associated with said second processing node;

a first node wrapper for coupling said a first core processor of said first processing node to said first memory and to said computational elements;

15 a second node wrapper for coupling said a second core processor of said second processing node to said second memory and to said computational elements; and

means for interconnecting said computational elements and said first and second processing nodes in a ring topology bus, to define a selected task to achieve a desired functionality.

20

34. (Original) The integrated circuit of claim 33 further comprising means for temporally adapting said second node and said computational elements to perform a selected function.

25 35. (Original) The integrated circuit of claim 34 wherein said temporal means further comprises executable code defining said selected function stored in at least said first memory.

30 36. (Original) The integrated circuit of claim 35 wherein said executable code is downloaded from the Internet by said first processing node.

S/N 10/765,556

37. (Original) The integrated circuit of claim 36 wherein said executable code comprises operating system code.

38. (Original) The integrated circuit of claim 37 wherein said first processing node
5 initiates the temporal adaptation of said computational elements and said second processing node to perform said selected function.

39. (Original) The integrated circuit of claim 33 wherein said computational elements include a plurality of arithmetic nodes, a plurality of bit-manipulation nodes and a
10 plurality of finite state machine nodes.

40. (Original) The integrated circuit of claim 33 further comprises a plurality of said second processing nodes each of which is coupled to said first processing node and computational elements by said interconnecting means.

15

41. (Currently amended) An adaptive computing engine comprising:

a controller node having: a core processor for executing operating system code;

a memory for storing operating system executable code;

means for transferring operating system executable code and data from said

20 memory to said core processor;

a plurality of computational elements coupled to each other through a ring topology bus and each of them adapted to perform a selected function, at least one of said

computational elements having: a RISC processor for executing code; a memory for storing executable code; means for transferring executable code and data from said

25 memory to said RISC processor; and

a temporal interconnecting matrix coupling said controller node to said plurality of computational elements to perform a user selected function.

S/N 10/765,556

42. (Currently amended) An adaptive computing engine having a plurality of computational elements and a temporal interconnecting matrix for connecting said computational elements, said adaptive computing engine comprising: a controller node for adapting said computational elements in response to perform a selected function, said
5 controller node having: a core processor for executing operating system code;
a memory for storing operating system executable code; means for transferring operating system executable code and data from said memory to said core processor;
a set of registers associated with said core processor; a node wrapper, coupled to said core processor, for receiving an input stream from an external source, said input
10 stream having configuration information and executable code and passing said information to said core processor; and
means for accessing said set of registers; an interrupt controller for detecting interrupt conditions from said node wrapper and internally; and
an input port and an output port, said input and output ports for coupling said
15 adaptive computing engine into a ring topology network.

43. (Original) The adaptive computing engine of claim 42 further comprising means for accessing said core processor and said memory to debug error conditions.

20 44. (Original) The adaptive computing engine of claim 42 further comprising means for handling node-to-node communication.

45. (Original) The adaptive computing engine of claim 42 further comprising executable code for controlling the temporal adaptation of said computation elements in response to
25 configuration information.

46. (Original) The adaptive computing engine of claim 45 further comprising means for controlling the initiation of operation of said computational element upon reset or power
on.

30

S/N 10/765,556

47. (Original) The adaptive computing engine of claim 46 further comprising a programmable scalar node having: a core processor for executing instructions;

an instruction memory for storing said instructions;

a data memory;

5 means for transferring instructions from instructions memory to said core processor and for transferring data to said core processor from said data memory;

a set of registers associated with said core processor; and

a node wrapper, coupled to said core processor, for receiving an input stream from controller node, said input stream having configuration information;

10 means for accessing said set of registers; and

an interrupt controller for detecting interrupt conditions.

48. (Original) The adaptive computing engine of claim 47 further comprising: a data cache; and an instruction cache.